Meets Specifications

Dear Student,

ETL

Thank you for the extremely extraordinary effort you have put into this project. I'm impressed!
I must tell you that this is one of the most well-formatted projects I ever happen to review.
We look forward to no less than this great submission for the coming projects.
Good luck and happy learning.

Friendly Note:
Stay safe and take care of yourself and all your beloved ones in the current circumstances :)

The script, etl.py, runs in the terminal without errors. The script reads song_data and load_data from S3, transforms them to create five different tables, and writes them to partitioned parquet files in table directories on S3.

Each of the five tables are written to parquet files in a separate analytics directory on S3. Each table has its own folder within the directory. Songs table files are partitioned by year and then artist. Time table files are partitioned by year and month. Songplays table files are partitioned by year and month.

Each table includes the right columns and data types. Duplicates are addressed where appropriate.
You did exactly as requested by my fellow previous reviewer, thank you for implementing this functionality! :)

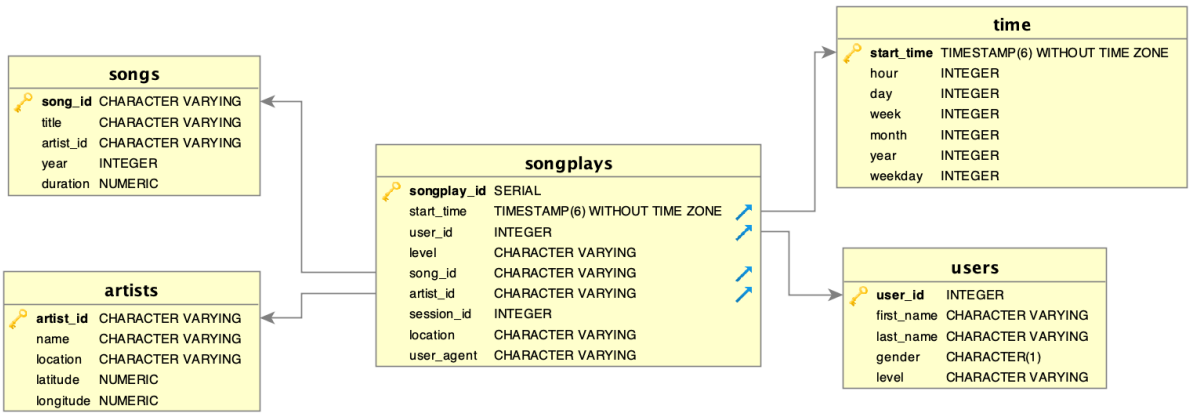
Code Quality

The README file includes a summary of the project, how to run the Python scripts, and an explanation of the files in the repository. Comments are used effectively and each function has a docstring.
Outstanding **README** file, you did exactly as requested by the rubric. Thank you for implementing that :)

Here are some general tips to follow whenever creating a **README** file for this project or for similar ones:

A nice README is a great way to showcase your project to potential employers. Following sections provide better readability:

- Introduction - purpose of project; in this case, what is Sparkify, how this project is going to help it.
- Database schema design and ETL process
- Files in repository
- How to run the python scripts
- Add a screenshot or an image (ER Diagram) showing how the fact and dimension tables are connected. Example:



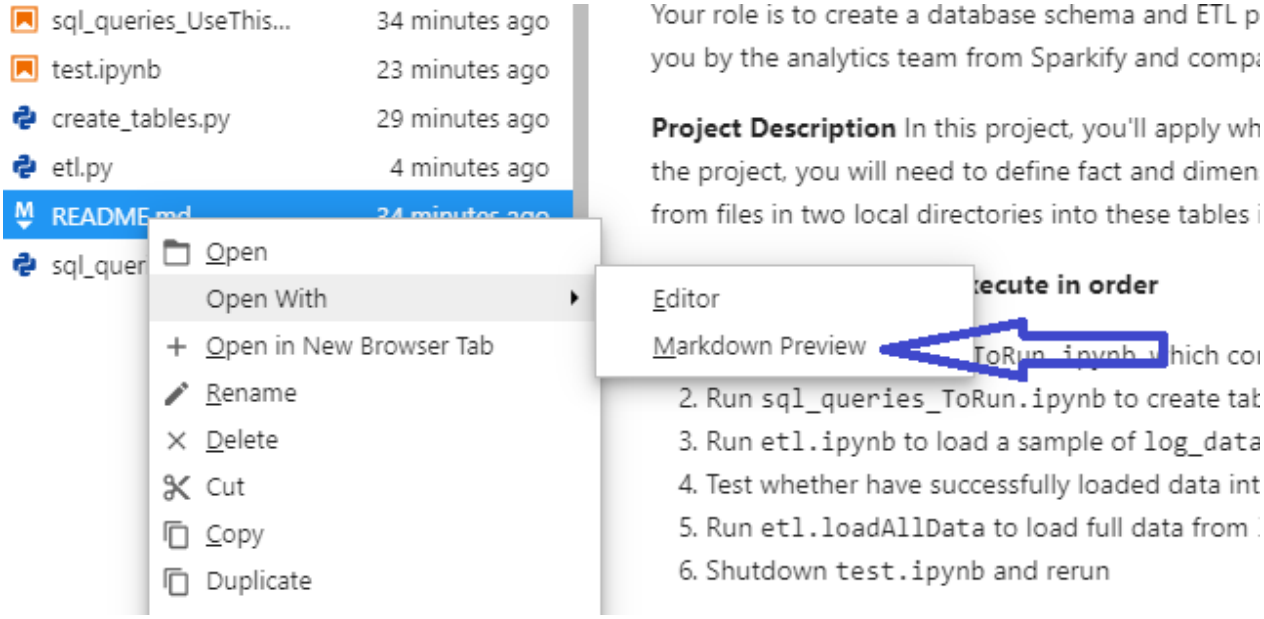
- Properly use the markdown language in your README, for headings, lists, subsections.
  Refer: https://guides.github.com/features/mastering-markdown/
- You can use an online markdown editor like https://dillinger.io/
- Refer good READMEs on web:

https://github.com/matiassingers/awesome-readme

https://bulldogjob.com/news/449-how-to-write-a-good-readme-for-your-github-project

https://medium.com/@meakaakka/a-beginners-guide-to-writing-a-kickass-readme-7ac01da88ab3

- You can preview your README in your Udacity workplace as below:



Scripts have an intuitive, easy-to-follow structure with code separated into logical functions. Naming for variables and functions follows the PEP8 style guidelines.

- Your project code is clean and for the most part follows the PEP8 style guidelines.
- I can clearly see your code structured into logical functions. Your function names clearly specify what your code is going to do. Great effort here!